On the Continuity and Origin of Identity in Distributed Ledgers: Learning from Russell's Paradox

Parra Moyano, José

Abstract: This article studies the origin and continuity of the identity of the entities inscribed in a distributed ledger. Specifically, it focuses on the differences between the identities of the entities that exist in a distributed ledger and those of the entities that exist outside the ledger but must be represented in the ledger in order to interact with it. It suggests that a distributed ledger that contains representations of entities that exist outside the ledger can yield a continuum of interconnected existing and past identities that is constantly redefined to represent new conceptual entities. This continuum can be understood as a metasortal—or a sortal of sortals—that resembles the mathematical structure of a set of sets. Further, the article presents the dilemma that arises when representing the identities of entities in a distributed ledger, and it draws an analogy between this dilemma and Russell's Paradox.

DOI: https://doi.org/10.1111/meta.12272

Originally published at:
Parra Moyano, José (2017). On the Continuity and Origin of Identity in Distributed Ledgers: Learning from Russell's Paradox. Metaphilosophy, 48(5):687-697.
DOI: https://doi.org/10.1111/meta.12272

ON THE CONTINUITY AND ORIGIN OF IDENTITY IN DISTRIBUTED LEDGERS: LEARNING FROM RUSSELL'S PARADOX

JOSÉ PARRA MOYANO
University of Zurich
jose.parramoyano@business.uzh.ch

Abstract: This article studies the origin and continuity of the identity of the entities inscribed in a distributed ledger. Specifically, it focuses on the differences between the identities of the entities that exist in a distributed ledger and those of the entities that exist outside the ledger but must be represented in the ledger in order to interact with it. This study suggests that a distributed ledger that contains representations of entities that exist outside the ledger can yield a continuum of interconnected existing and past identities that is constantly redefined to represent new conceptual entities. This continuum can be understood as a meta-sortal—or a sortal of sortals—that resembles the mathematical structure of a set of sets. Further, this article presents the dilemma that arises when representing the identities of entities in a distributed ledger, and draws an analogy between this dilemma and Russell's Paradox.

Keywords: Blockchain, DLT, identity, Russell's Paradox, sortal.

1. Introduction

The idea of smart contracts, which facilitate, verify, or enforce predefined clauses whenever a set of conditions is given, was introduced by Szabo (1997). Thanks to Nakamoto's seminal paper (2008), smart contracts have become more useful, and thus more relevant. As stated by Buterin (2015), the distributed ledger technology (DLT) described by Nakamoto (2008) can be extended to host and execute smart contracts in a decentralized and autonomous way. One characteristic of distributed ledgers is that the digital tokens that operate in the system can be traced back in time by any node in the ledger. These tokens are divisible and every part of a token can also be traced along the distributed ledger.

Many researchers and practitioners are currently working on the application of DLT to solving existing problems and inefficiencies in information systems (IS). Some of these applications require entities (such as a fiat currency, a property title, an identity certificate, a property title, etc.) that exist outside a distributed ledger to interact with smart contracts within that ledger. External entities' interaction with a distributed ledger requires these entities to be represented by a digital token within the ledger. This implies that the identity of external entities needs to be inscribed by an instance in the distributed ledger. The study of the identity of entities in distributed ledgers enriches the discussion with regard to the continuity and the origin of identity. Moreover, studying the concept of the identity of the entities in a distributed ledger can help us to better understand the limitations of the different types of distributed ledgers that currently exist, and to propose more accurate applications of DLT and define further dimensions of the technology.

Part 2 of this article provides an introduction to DLT and presents the concept of identity in distributed ledgers, explaining the characteristics of tokens in a ledger with a focus on the technical differences between those entities that are created in a distributed ledger and those that exist outside that ledger. Part 3 analyzes the concept of the identity of the tokens of a distributed ledger, studies that identity in terms of these tokens' spatiotemporal and qualitative identity careers, and presents the idea of DLT as a meta-sortal—a sortal that itself contains sortals and that helps to define a continuum of interrelated identities of entities within a distributed ledger. Part 4 indicates the contradictions that occur when trying to inscribe the identity of external entities into a distributed ledger and draws an analogy between these contradictions and Russell's Paradox. Part 5 concludes.

2. Identity Representations in Distributed Ledgers

DLT facilitates a distributed, transactional database in which globally distributed nodes are linked by a peer-to-peer (P2P) communication network with a layer of protocol messages that enables them to communicate. Glaser (2017) points out that the main purpose of DLT is to provide a database that operates validated, immutable transactions, such that historical transactions at a single node result in invalid states should any node try to rewrite or alter the historical record of transactions. Smart contracts are autonomous, interacting pieces of code that can be built into a distributed ledger. It is important to note that smart contracts need to be triggered by a third party and cannot trigger themselves autonomously. Once a smart contract is executed, it starts to autonomously interact with other pieces of code.

DLT also enables the transfer of tokens within a system. As described by Glaser (2017), the mere technical creation of a token in a distributed ledger does not necessarily attach any value to that token. Glaser (2017) defines two types of distributed ledger depending on the value of the tokens inscribed in the ledger. Type 1 distributed ledgers are those that function properly if the nodes can rely on the internally generated tokens to interoperate. Type 2 distributed ledgers are those that need frequent interaction with external services in order to bind value to the tokens in the system.

It is therefore essential to differentiate between tokens that have value from the moment of their creation in a distributed ledger system and those that do not. An example of a token that has value from the moment of its creation is a bitcoin, which only starts to exist when it is created as a token in a distributed ledger and needs no instance to grant it value within that ledger. An example of a token that contains no value at the point of its creation is a token that is created to work as a digital representation of an external entity that does not, itself, exist in the distributed ledger (e.g., a property title, a fiat currency, a passport, a quantity of gold, etc.). An external entity that is meant to interact with a smart contract in order to benefit from the characteristics of DLT needs to be represented in the distributed ledger by a token. Therefore, a token that is to represents an external entity in a distributed ledger needs to be granted an identity in that ledger such that every node in the network is aware that, from a certain point in time onward, that specific token represents an external entity inside the ledger. Granting a token an identity automatically attaches value to it, since this token now represents the external entity inside the ledger. Moreover, granting an entity a digital identity within a distributed ledger ensures that a representation of this entity can operate with a smart contract, and benefit from the characteristics of DLT-stored data—that is to say, reliability, confidentiality, integrity, availability, and non-repudiation. Tokens in a distributed ledger have a unique and traceable identity. This enables the nodes in the ledger to observe any entity in the ledger, and any part of this entity across time. The action of granting a token an identity needs to be conducted by an instance that can write data in the distributed ledger and that can verify the status of the physical entity outside the ledger[1].

3. Continuity of Identity in Distributed Ledgers

The concept of continuity of identity has been studied in many contexts. Hirsch (1982) states that, considering any object, it is always possible to trace any indefinite number of qualitatively and spatiotemporally continuous or discrete space-time paths, which connect the whole object at one time to any of its parts at a later time; and also

[1] This process of granting identity to tokens that represent entities external to the ledger—along with the consequent ability to trace entities, and parts of entities, across time—is illustrated by Bitcoin's "colored coins" (see Assia et al. 2012)

that, considering any part of any object, it will always be possible to trace an indefinite number of qualitatively continuous space-time paths, which connect that part of the object at one time to some other part of the object at another time. However, before the introduction of distributed ledgers, "persisting" objects, which correspond to these traceable paths, generally did not exist as ordinarily conceived, since—in terms of our ordinary conception—there exists no persisting object that combines the various stages of the *whole* object with the various stages of that object's *parts*, or that combines the various stages of *one* part of an object with the various stages of *other* parts. Hirsch (1982) considers that if these objects did exist, we would be able to observe a menagerie of pseudo-careers made up of the scattered stages of the career of the whole and the careers of its parts. The reliance such on continuity considerations for objects that only exist outside a distributed ledger would consequently yield the most "*drastic and bizarre identity deviations"* (Hirsh, 1982).

Continuity of identity can also be illustrated by the wax argument introduced by Descartes (1641) in his *Meditations on First Philosophy*. The wax argument is a thought experiment that analyzes the essential properties of bodies. In this thought experiment, Descartes argues as follows:

> *"Let us take, for example, this piece of wax: it has been taken quite freshly from the hive, and it has not yet lost the sweetness of the honey which it contains; it still retains somewhat of the odour of the flowers from which it has been culled; its colour, its figure, its size are apparent; it is hard, cold, easily handled, and if you strike it with the finger, it will emit a sound. Finally, all the things which are requisite to cause us distinctly to recognize a body, are met with in it. But notice that while I speak and approach the fire what remained of the taste is exhaled, the smell evaporates, the colour alters, the figure is destroyed, the size increases, it becomes liquid, it heats, scarcely can one handle it, and when one strikes it, no sound is emitted. Does the same wax remain after this change? We must confess that it remains; none would judge otherwise."*

Descartes conclude that the wax could be extended in ways that he could not accurately imagine.

Now, and thanks to the characteristics of DLT, the *drastic and bizarre identity deviations* mentioned by Hirsh (1982) can indeed occur. Similarly, our imagination can suddenly conceive of more extensions of Descartes' wax, should a representation of that piece of wax be inscribed in a distributed ledger. With DLT it is literally possible to trace any divisible part of a token. This enables the nodes of the ledger to observe a) when an entity, as the sum of many smaller entities, comes to life, b) how this entity gets divided, traded, and recomposed across time, and c) when this entity gets reassembled with other entities or parts of entities to give rise to a new entity.

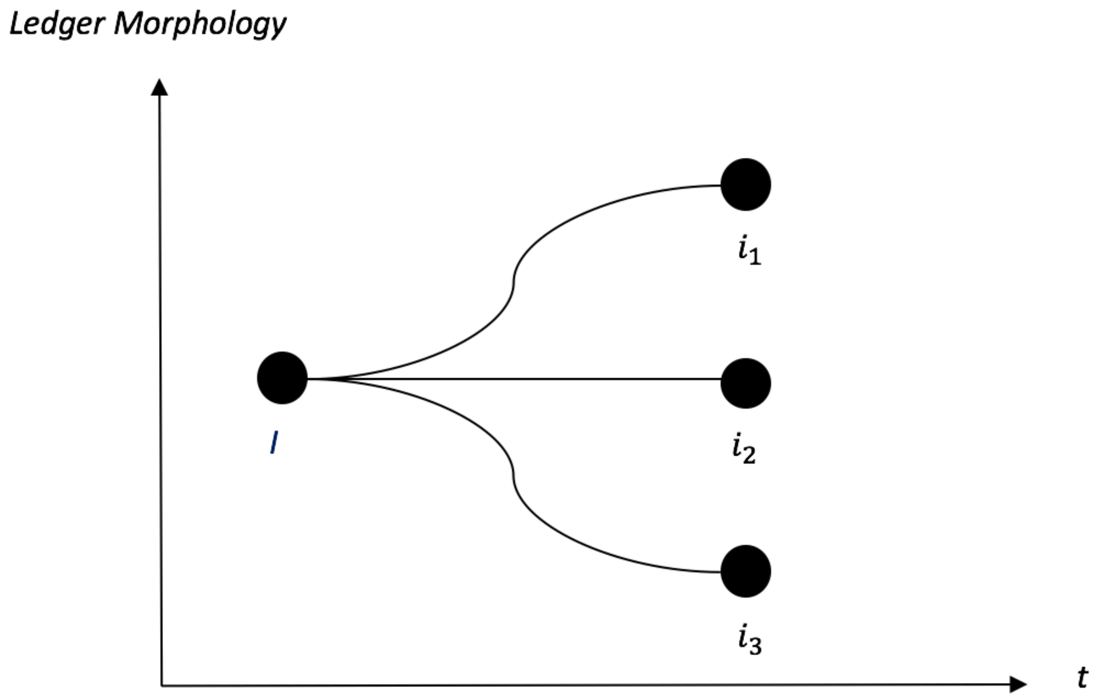

Figure 1. Representation of an entity I and its parts on the ledger

Assuming that an entity has an identity, I, inscribed in a distributed ledger from a certain time point, t, onward allows any node in the ledger to trace I, and any change in I, for as long as the distributed ledger exists or for as long as I exists within the ledger. Assuming that I is composed of n parts, such that $I = i_1 + i_2 + \cdots + i_n$, any node can trace if I is, or has ever been, subdivided, recomposed, or altered in time. This means that, thanks to DLT, any node can always trace each part of an entity I. Moreover, DLT allows the *tracing* of any divisible part of the identity I at any time period t', as well as any combination of I as a whole with another entity I', or with any part of another entity I'.

Figure 1 represents an entity I in a ledger and how this entity can be subdivided into many parts that can be separately spread along the ledger, generating a new ledger morphology.

This concept might be easier to understand by means of an example. Assuming that a block of wood, W, is inscribed in a distributed ledger at time t, any observer of that ledger can observe W. W can then be subdivided in n parts, such that $W = w_1 + w_2 + \cdots + w_n$. The block of wood W can be halved, such that one half can be shaped to form a chair C, and the other half can be shaped to form a desk D. Metadata can be written in a distributed ledger by an instance that is entitled to do so. Writing metadata in a distributed ledger would allow us to record the abovementioned subdivision and shaping. At a later stage t' any node in the ledger can identify a chair C, which is composed of half of the original wood block W, and a desk D, composed of half of the original wood block W.

This results in a situation in which each of the newly generated entities (the chair and the desk) has a new identity, such that $C = c_1 + c_2 + \cdots + c_m$ and $D = d_1 + d_2 + \cdots + d_p$. However, the nodes in the ledger can trace C and D back in time, and realize that C and D were formed from parts of W, such that $W = C + D$. Moreover, any observer of the ledger can identify from which specific parts of W each of the new entities C and D are formed, such that all the components of C are formed from some components of W and all the components of D are composed of some other, different, components of W, such that—for example—$C = c_1 + c_2 + \cdots + c_m = w_1 + w_2$ and $D = d_1 + d_2 + \cdots + d_p = w_3 + \cdots + w_n$.

This concept makes it possible to add up additional degrees of complication. Assuming that at time t'' the chair is crushed to form particles of sawdust, every particle of sawdust can be identified and traced back as having previously belonged to a specific part of the chair and, previously, to a specific part of the wood block. If this sawdust would later be resembled to form another entity, any observer of the ledger could trace back that entity's origin to the very first point in time at which the first component of the analyzed entity was inscribed in the ledger. This implies that, in a distributed ledger, identity can be traced from the current time period to any past time period in a very specific way, which generates a continuum of interconnected existing and past identities that is constantly redefined to represent new conceptual entities.

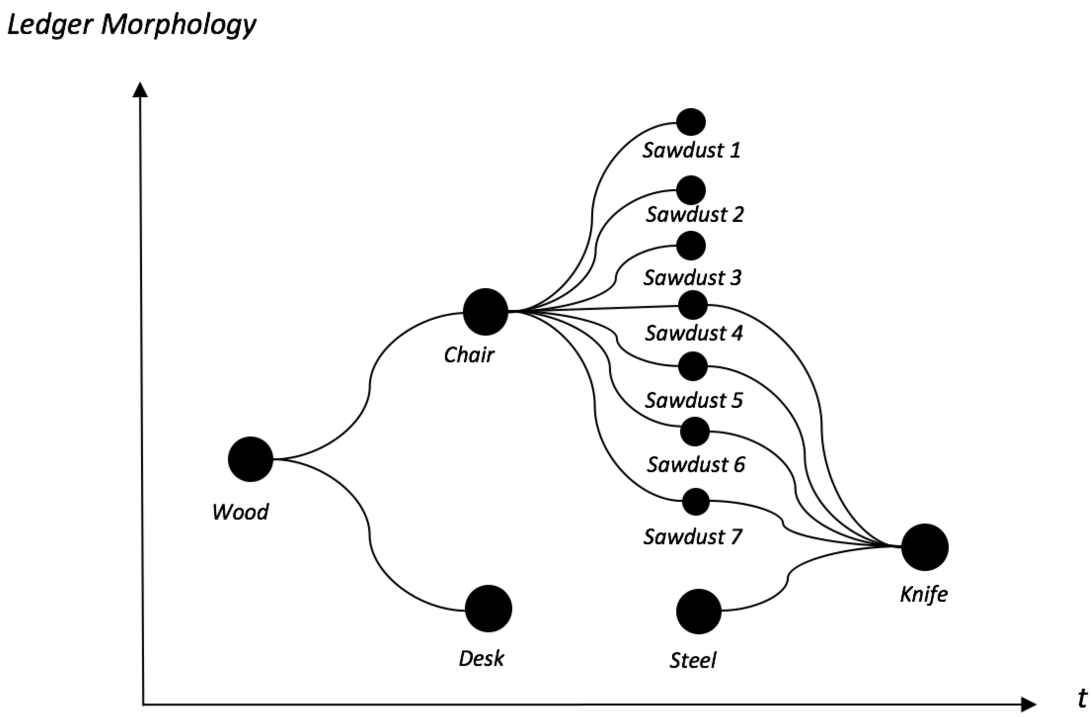

Figure 2. Example of different entities and their ramifications in the ledger

Figure 2 illustrates how the identity of any entity in a distributed ledger can be traced back in time, observing any combination with other entities, demonstrating how some sawdust particles could be reassembled with another entity—for example, a piece of steel—to form a new entity, such as a knife.

The consequence of this infinite and unalterable traceability is that all entities that have their identity inscribed in a distributed ledger can yield new entities that result from the combination of the identities of previously existing entities, such that the reality in the ledger can be understood as a continuous combination of identities.

This might also have implications when it comes to the concept of the "sortal". Taking our definition from the *Stanford Encyclopedia of Philosophy*, a sortal is a concept that:

a) tells us what the essence of a thing is

b) tells us how to count things of that kind, which requires knowing which things are different and which are the same, and

c) tells us when something continues to exist, and when it goes out of existence

A distributed ledger (understood as the transactional record of the combinations of its entities, or its graphical representation) fulfills these three requirements:

a) The ledger tells us what the essence of a thing is, by clearly defining from which parts each entity is composed, and by defining the tokens that represent the entity as a whole.

b) The ledger helps us to count and compare the parts of one or many things, such that equal or similar things can be counted together and different things can be differentiated due to their specific composition.

c) By inscribing metadata in the ledger, it is clear when an entity comes into and goes out of existence. In fact, the precise point in time that these events occur is registered in the ledger and cannot be altered by any node.

Since different entities can be inscribed in a distributed ledger, a sortal can also be inscribed within a distributed ledger. Therefore, a distributed ledger can be seen as a meta-sortal that can help the nodes of the ledger to observe and keep track of the origin and continuity of the identity of the entities inscribed in the ledger. A meta-sortal can be understood as a sortal that itself contains sortals. This concept of the meta-sortal resembles the concept of the "set of sets", which played a central role in the development of Russell's Paradox. Having defined the concept of the meta-sortal, a distributed ledger can be seen as a construct in which many other constructs can be contained. This is de facto one characteristic of a distributed ledger, which can be a construction in which all the elements of an economy (currencies, enterprises, decentralized organizations, consumers, platforms, etc.) can be contained. Drawing an analogy between distributed ledgers and sets allows us to make the distinction between open and closed ledgers, which resembles the distinction between open and closed *sets*. As long as a distributed ledger needs to interact with entities outside of itself, this ledger is open, in the sense that it interacts with entities outside the ledger. Similarly, as long as a distributed ledger does *not* interact with entities outside of itself this ledger is closed. Open and closed sets differ, as do open and closed ledgers, with regard to the properties that they have at their borders.

4. Identity Conflicts in Distributed Ledgers

Type 1 ledgers defined by Glazer (2017), which I will here refer to as "closed distributed ledgers", are those in which nodes can rely on the internally generated tokens to interoperate. In ledgers of Type 1, the tokens in the system need no instance to grant them an identity since they are stand-alone objects with an intrinsic value. Type 2 ledgers defined by Glazer (2017), which I will refer to here as "open distributed ledgers", are those that need frequent interaction with external services in order to bind value to the system. In ledgers of Type 2, tokens are created without any value in the system, and require a validating instance that grants them an identity if they are to have any value. As mentioned in Part 2, this validating instance also needs to be able to write data in the distributed ledger and to verify the status of the physical entity outside the distributed ledger.

In distributed ledgers of Type 1 the whole career of the entity is inscribed in the ledger. However, the careers of the entities that exist outside the ledger before their identity has been inscribed in the distributed ledger by a validating instance cannot be fully observed by the nodes of the distributed ledger (since this instance exists outside the distributed ledger before having an identity in the ledger). Hence, in ledgers of Type 2, nodes need to rely on a validating instance to inscribe the identity of external entities in the distributed ledger. The case of bitcoin clearly illustrates one example of a distributed ledger of Type 1. In the bitcoin blockchain there is no need of a trusted third party that grants the bitcoins with value or with an identity, since a bitcoin is a stand-alone entity whose career can be clearly observed by any node. Hence, the creation of the (in this case digital) entity is automatically accompanied by the inscription of its identity in the distributed ledger, such that there is no need of an instance to carry out this inscription.

Distributed ledgers of Type 2 require that interaction takes place between the ledger and a validating instance that grants the tokens of the ledger an identity. This generates a conflict due to the fact that such a validating instance can only inscribe in the distributed ledger the identity of those entities that cannot be created in the ledger and that need a token to represent them, since otherwise the validating instance would be granting a token an identity that that token already has. In such a case, the validating instance would be redundant. However, since such a validating instance cannot be created within a distributed ledger (because it must interact with the ecosystem that exists outside the ledger), if it is to be able to participate in the ledger, the instance will have to verify itself, which leads to a logical contradiction.

This problem very much resembles the structure of Russell's Paradox (Russell 1918), which he used to criticize Set Theory as defined by Cantor; the Paradox states that:

> "*The barber is the one who shaves all those, and those only, who do not shave themselves. The question is, does the barber shave himself?*"

Reformulating this paradox in terms of distributed ledgers of Type 2 would yield the following paradox:

> "*The inscribing instance is the instance that grants identity in the distributed ledger to all those, and those only, who cannot grant themselves their own identity in the ledger. The question is, does the inscribing instance grant itself its own identity in the distributed ledger?*"

Expressed in terms of first order logic, the paradox of the inscribing instance would be defined as follows:

$$\forall \boldsymbol{x} \quad \boldsymbol{grant\ identity}(\boldsymbol{x}, \boldsymbol{instance}) \Leftrightarrow \neg \boldsymbol{grant\ identity}(\boldsymbol{x}, \boldsymbol{x}), \qquad (\mathbf{1})$$

where $grant\ identity(x, y)$ means "x is granted an identity by y", such that Equation (1) could be read as "every entity is granted an identity by a verifying instance if and only if this entity cannot grant itself an identity". This would imply the contradiction stated in Equation (2), which states that the verifying instance grants itself an identity if and only if it does *not* grant itself an identity:

$$\boldsymbol{grant\ identity}(\boldsymbol{instance}, \boldsymbol{instance}) \Leftrightarrow \neg \boldsymbol{grant\ identity}(\boldsymbol{instance}, \boldsymbol{instance}). \qquad (\mathbf{2})$$

Hence, there is a distinction between distributed ledgers that are independent of a verifying instance, and are therefore "normal" in the sense of the smooth but trustless cooperation that they enable, and distributed ledgers that need a verifying instance to inscribe the identity of the entities that form them, and that are therefore "singular" distributed ledgers in the sense that they need a certain ex ante agreement between the participants of the ledger in order to inscribe the identities of the entities that form the ledger. All closed distributed ledgers are normal ledgers, whereas open distributed ledgers might be singular ledgers, depending on the characteristics of the system.

5. Conclusion

This article has presented how an entity inscribed in a distributed ledger can be traced, together with its parts, across time, which enables us to understand the identity of an entity as a dynamic, interconnected continuum of prior entities' identities that gives rise to a new set of identities. This characteristic allows us to understand a distributed ledger as a meta-sortal, which is a sortal of sortals, and hence as a construct in which other constructs can be contained. Furthermore, this article differentiates between open and closed ledgers, resembling the differences between open and closed sets, and describes the dilemma that arises in distributed ledgers of Type 2, drawing an analogy between this dilemma and Russell's Paradox.

The definition of a distributed ledger as a meta-sortal, and the description of the dilemma that occurs in ledgers of Type 2, serve a double purpose. First, they can help us to better understand the nature of distributed ledgers as a construct of constructs (or as a set of sets). Second, they can contribute to further developing definitions of smaller theoretical bodies within the concept of the distributed ledger. DLT can establish the foundations for new types of economic settings and information systems. However, at the current stage in the development of DLT, pointing out dilemmas and new concepts with regard to DLT, such as those exposed here, might help researchers and practitioners to advance the development of, to propose a more accurate application of, and to define the further dimensions of DLT.

References

Assia, Yoni, Vitalik Buterin, Liorhaki Lior Hakim, Meni Rosenfeld, and Lev Rotem. 2016. *Colored Coins White Paper*. Online: https://goo.gl/TVhlfD (retrieved 14.05.2017).

Batlin, Alex. 2015. *Crypto 2.0 Musings - Digital Identity on Blockchain*. Online: www.linkedin.com/pulse/crypto-20-musings-digital-identity-blockchain-alex-batlin. (retrieved 13.09.2016).

Buterin, Vitalik. 2015. *A Next Generation Smart Contract and Decentralized Application Platform*. *Ethereum White Paper*. Online: https://github.com/ethereum/wiki/wiki/White-Paper (retrieved 01.09.2016).

Descartes, René. 1641. *Meditations on First Philosophy*. Translated by Elizabeth S. Haldane. Cambridge, UK: Cambridge University Press.

Glaser, Florian. 2017. *Pervasive Decentralisation of Digital Infrastructures: A Framework for Blockchain enabled System and Use Case Analysis*. Proceedings of the 50th Hawaii International Conference on System Sciences.

Hirsch, Eli. 1982. *The concept of Identity*. Oxford, United Kingdom: Oxford University Press.

Nakamoto, S. 2008. *Bitcoin: A Peer-to-Peer Electronic Cash System*. Online: https://bitcoin.org/bitcoin.pdf (retrieved 15.09.2016).

Oxford English Dictionary. 2017. Definition of "sortal". Online: https://en.oxforddictionaries.com/definition/us/sortal (retrieved 19.09.2016).

Probst, Laurent, Laurent Frideres, Benoît Cambier, and Christian Martinez-Diaz. 2016.

Blockchain applications and services. Luxembourg: Business Innovation Observatory, European Commission.

Russell, Bertrand. 1918. *Logic and Knowledge*. New York: Capricorn Books.

Stanford Encyclopedia of Philosophy. Definition of “sortal”. Online: http://plato.stanford.edu/entries/sortals/. (retrieved 28.09.2016) (Stanford, CA: Metaphysics Research Lab, Stanford University).

Szabo, Nick. 1997. “Smart Contracts: Formalizing and Securing Relationships on Public Networks”. *First Monday*, Volume 2, Number 9.